SQL TIP

Manoj Kumar

Row_Number() vs Rank() vs Dense_Rank()

https://www.linkedin.com/in/mk-analytics

Manoj Kumar

ROW_NUMBER()

The ROW_NUMBER() function assigns a consecutive ranking to each row within a result set, ordered by a specific column or set of columns. For example, if you use ROW_NUMBER() to rank a set of rows by salary, it will assign a ranking to each row based on the salary values, with the highest salary receiving a ranking of 1. Here is an example of how to use the ROW_NUMBER() function:

```
--Here is an example of how to use the ROW_NUMBER() function:

SELECT
  row_number() OVER (ORDER BY salary DESC) AS rank,
  name,
  salary
FROM employees
ORDER BY salary DESC;
```

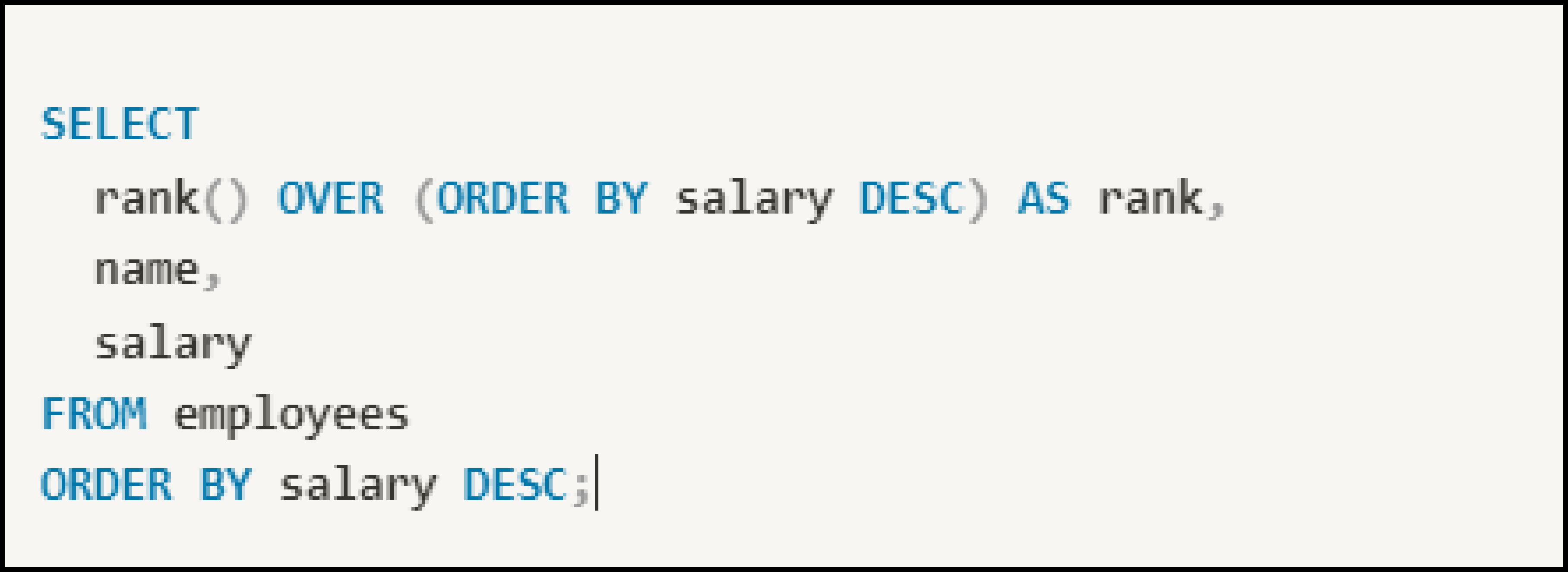
```
--This query will return the following results:

rank | name | salary
-----|------|--------
1 | John Smith | 100000
2 | Jane Doe | 90000
3 | Peter Jones | 80000
```

As you can see, the ROW_NUMBER() function has assigned a consecutive ranking to each row, starting with 1 for the row with the highest salary, and continuing down to the row with the lowest salary.

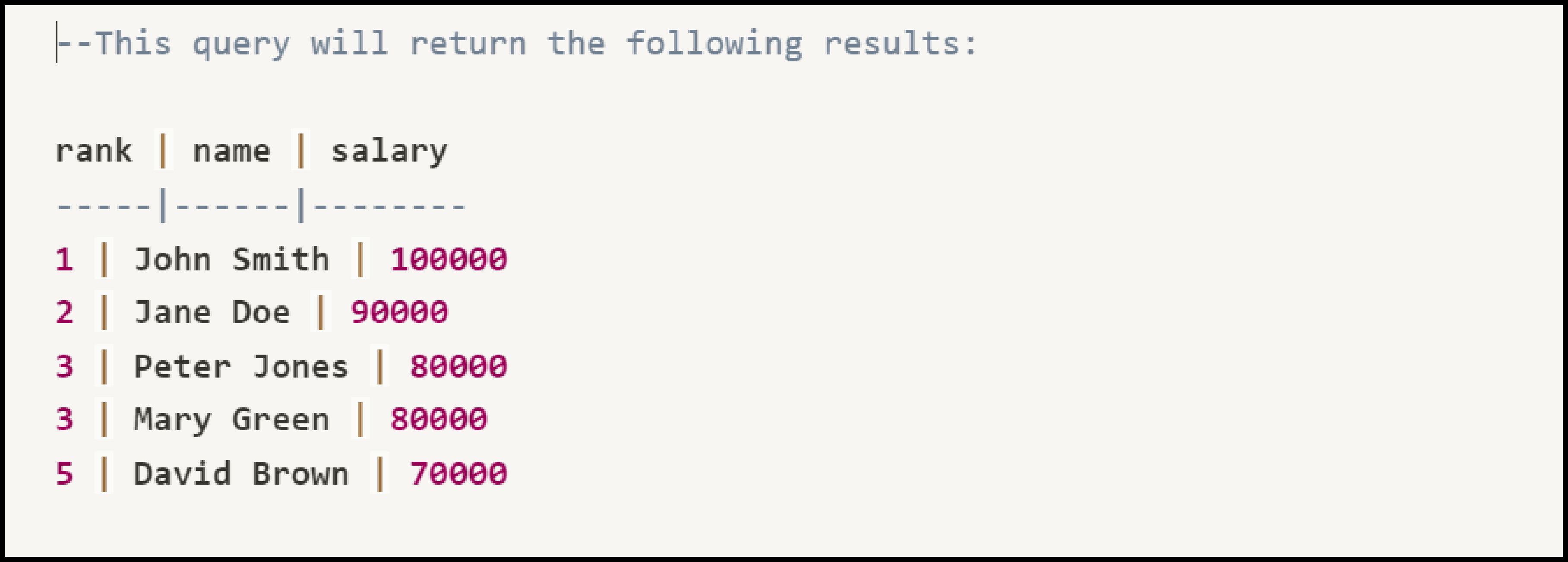
Manoj Kumar

RANK()

The RANK() function assigns a ranking to each row within a result set, similar to ROW_NUMBER(). However, it assigns the same ranking to rows that have the same value in the order by column or set of columns. For example, if you use RANK() to rank a set of rows by salary, it will assign the same ranking to rows with the same salary value. This can result in gaps in the ranking values, as shown in the following example:



```sql
SELECT
  rank() OVER (ORDER BY salary DESC) AS rank,
  name,
  salary
FROM employees
ORDER BY salary DESC;
```



```
--This query will return the following results:

rank | name | salary
-----|------|--------
1 | John Smith | 100000
2 | Jane Doe | 90000
3 | Peter Jones | 80000
3 | Mary Green | 80000
5 | David Brown | 70000
```

As you can see, the RANK() function has assigned the same ranking of 3 to both Peter Jones and Mary Green. This is because they both have the same salary value. And Rank 5 is assigned to David Brown and skips Rank 4.

https://www.linkedin.com/in/mk-analytics/

DENSE_RANK()

Manoj Kumar

The DENSE_RANK() function is similar to RANK(), but it does not leave gaps in the ranking values. It assigns the same ranking to rows with the same value in the order by column or set of columns, but it also assigns the next highest ranking to the next row, regardless of the value in the ordered column.

```
--Here is an example of how to use the DENSE_RANK() function:

SELECT
  dense_rank() OVER (ORDER BY salary DESC) AS rank,
  name,
  salary
FROM employees
ORDER BY salary DESC;
```

```
--This query will return the following results:

rank | name | salary
-----|------|--------
1 | John Smith | 100000
2 | Jane Doe | 90000
3 | Peter Jones | 80000
3 | Mary Green | 80000
4 | David Brown | 70000
```

As you can see, the DENSE_RANK() function has assigned the same ranking of 3 to both Peter Jones and Mary Green, but it has also assigned a ranking of 4 to David Brown.
This is because the DENSE_RANK() function does not leave gaps in the ranking values.

https://www.linkedin.com/in/mk-analytics/

Manoj Kumar

Looking for Real-World Experience in Data Analytics/BI?

IM me on LinkedIn to know more

Or

click here to book a call

https://www.linkedin.com/in/mk-analytics